



STIC EIC 2100 12/826

Search Request Form 49

Today's Date:

5/12/2004

What date would you like to use to limit the search?

Priority Date: 10/12/2000

Other:

Name Kuen S. Lu

AU 2177 Examiner # 79991

Room # PK2 4A32 Phone 305-4894

Serial # 09/973,195

Format for Search Results (Circle One):

PAPER

DISK

EMAIL

Where have you searched so far?

USP

DWPI

EPO

JPO

ACM

IBM TDB

IEEE

INSPEC

SPI

Other google

Is this a "Fast & Focused" Search Request? (Circle One) YES NO

A "Fast & Focused" Search is completed in 2-3 hours (maximum). The search must be on a very specific topic and meet certain criteria. The criteria are posted in EIC2100 and on the EIC2100 NPL Web Page at <http://ptoweb/patents/stic/stic-tc2100.htm>.

REZ complete by 5 pm 5/12/2004 THANKS

What is the topic, novelty, motivation, utility, or other specific details defining the desired focus of this search? Please include the concepts, synonyms, keywords, acronyms, definitions, strategies, and anything else that helps to describe the topic. Please attach a copy of the abstract, background, brief summary, pertinent claims and any citations of relevant art you have found.

receiving a query for a path of related objects, the query including information regarding

an object and relationships associated with the object;

analyzing the different relationships associated with objects in the system to select

objects based on said information included in the query; and

based on said analysis, selecting a path of related objects, one of the objects in the path

being the object indicated by the query.

*The query is for selecting "path of related objects".
The query statement has information of "an object and relationships associated with the object".*

STIC Searcher David Holloway

Phone 308-7784

Date picked up 5-12-04

Date Completed 5-12-04



*DTALOG 782⁹⁰
60 min*



STIC Search Report

EIC 2100

STIC Database Tracking Number 121826

TO: Kuen S Lu
Location: 4A32
Art Unit : 2177
Wednesday, May 12, 2004

Case Serial Number: 09/973195

From: David Holloway
Location: EIC 2100
PK2-4B30
Phone: 308-7794

david.holloway@uspto.gov

Search Notes

Dear Examiner Lu,

Attached please find your search results for above-referenced case.
Please contact me if you have any questions or would like a re-focused search.

David

PLZ
Keep
this
doc.
THANKS. A Lu 5/12/04



Set	Items	Description
S1	5031	OO OR OBJECT()ORIENT? OR OODB OR C()PLUS OR CPLUS OR ORDBMS OR ORDB
S2	941794	PATH? OR ROUTE? OR RELATED OR INHERIT? OR CHILD? OR PARENT? OR ANCESTOR? OR INSTANCE(2N)LOCATION?
S3	1170257	QUER? OR SEARCH? OR SEEK? OR FIND? OR LOCAT? OR RETRIEV?
S4	1832670	SETS OR SET OR BAGS OR CLASSES
S5	463847	OBJECT OR OBJECTS
S6	215	S1 AND S2 AND S3
S7	215	S6 AND S3
S8	84	S7 AND S4
S9	15	S7 AND IC=G06F-007?
S10	19297	S2(4N)S3
S11	18	S8 AND S10
S12	32	S9 OR S11
S13	32	IDPAT (sorted in duplicate/non-duplicate order)
S14	32	IDPAT (primary/non-duplicate records only)

File 347:JAPIO Nov 1976-2003/Dec(Updated 040402)
(c) 2004 JPO & JAPIO

File 350:Derwent WPIX 1963-2004/UD,UM &UP=200428
(c) 2004 Thomson Derwent

PLZ
Keep
this
set
of
doc
for
next
action
W Lu 5/13/04

14/5/10 (Item 10 from File: 350)
DIALOG(R) File 350:Derwent WPIX
(c) 2004 Thomson Derwent. All rts. reserv.

014979115 **Image available**
WPI Acc No: 2003-039629/200303
Related WPI Acc No: 2001-432376; 2001-625902; 2002-414155; 2003-039626;
2003-058087; 2003-220403; 2003-902144; 2004-060691; 2004-061065
XRPX Acc No: N03-030991

Database management system e.g. RDBMS for Internet URL directory systems,
has integrated multidimensional data aggregation module and relational
datastore between which bidirectional data flow occurs for exchanging
data

Patent Assignee: HYPERROLL ISRAEL LTD (HYPE-N)
Inventor: BAKALASH R; CASPI J; SHAKED G
Number of Countries: 001 Number of Patents: 001
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
US 20020129032	A1	20020912	US 2000514611	A	20000228	200303 B
			US 2000634748	A	20000809	
			US 2001796098	A	20010228	
			US 200136734	A	20011107	

Priority Applications (No Type Date): US 2001796098 A 20010228; US
2000514611 A 20000228; US 2000634748 A 20000809; US 200136734 A 20011107

Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
US 20020129032	A1		71	G06F-007/00	CIP of application US 2000514611 CIP of application US 2000634748 Cont of application US 2001796098 CIP of patent US 6385604

Abstract (Basic): US 20020129032 A1

NOVELTY - The DBMS has an integrated multidimensional data
aggregation module connected to a relational datastore. Bidirectional
data flow occurs between the relational datastore and the integrated
aggregation module such that the data stored in the relational
datastore is loaded into the aggregation module and the aggregated data
stored in the aggregation module is communicated to the relational
datastore.

USE - E.g. relational DBMS (RDBMS), object DBMS (ODBMS),
object-relational DBMS (ORDEMS) for multidimensional online
analytical processing systems (MOLAP), relational (OLAP) systems,
Internet URL directory systems, personalized online e-commerce shopping
systems, Internet based systems.

ADVANTAGE - Provides improved response time in handling queries
issued to the DBMS, thus enabling enterprise-wide centralized
aggregation. Minimizes the data handling operations in multi-hierarchy
data structures, and eliminates the need to wait for full aggregation
to be complete.

DESCRIPTION OF DRAWING(S) - The figure shows the schematic block
diagram of stand-alone aggregation server.

pp; 71 DwgNo 6B/22

Title Terms: DATABASE; MANAGEMENT; SYSTEM; DIRECTORY; SYSTEM; INTEGRATE;
MULTIDIMENSIONAL; DATA; AGGREGATE; MODULE; RELATED ; BIDIRECTIONAL; DATA
; FLOW; OCCUR; EXCHANGE; DATA

Derwent Class: T01

International Patent Class (Main): G06F-007/00

International Patent Class (Additional): G06F-017/00

File Segment: EPI

14/5/12 (Item 12 from file: 350)
DIALOG(R) File 350:Derwent WPIX
(c) 2004 Thomson Derwent. All rts. reserv.

014622757 **Image available**
WPI Acc No: 2002-443461/200247
XRPX Acc No: N02-349394

**Object determination method for object oriented computer system,
involves selecting path of related objects based on analysis of
different relationships associated with objects**
Patent Assignee: ABB AB (ALLM); ANDERSSON J (ANDE-I); PAULY T (PAUL-I);
RUDIN M (RUDI-I)
Inventor: ANDERSSON J; PAULY T; RUDIN M
Number of Countries: 004 Number of Patents: 004
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
US 20020046208	A1	20020418	US 2001973195	A	20011010	200247 B
SE 200103361	A	20020413	SE 20013361	A	20011010	200247
DE 10149693	A1	20020523	DE 1049693	A	20011009	200254
GB 2371884	A	20020807	GB 200025051	A	20001012	200259

Priority Applications (No Type Date): GB 200025051 A 20001012

Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
US 20020046208	A1		11	G06F-007/00	
SE 200103361	A			G06F-009/00	
DE 10149693	A1		12	G06F-017/30	
GB 2371884	A			G06F-017/30	

Abstract (Basic): US 20020046208 A1

NOVELTY - A **query** including information regarding an object and relationships associated with the object, is received for determining the **path** of the **related** objects. The **path** of the **related** objects is selected based on the analysis of different relationships associated with the objects in the system. One of the objects in the **path** is the object indicated in the **query**.

DETAILED DESCRIPTION - INDEPENDENT CLAIMS are included for the following:

- (1) Computer program for determining **path** of **related** objects to **find** object in computerized control system;
- (2) Data processing system; and
- (3) Computer data signal.

USE - For controlling e.g. manufacturing and process industry such as chemical plants, oil refineries, pulp and paper mills, steel mills, automated factories etc.

ADVANTAGE - The object is found based on analysis of contextual information, hence the **search** function is made faster.

DESCRIPTION OF DRAWING(S) - The figure shows the steps used to compute a **set** of **related** objects.

pp; 11 DwgNo 5/5

Title Terms: OBJECT; DETERMINE; METHOD; OBJECT; ORIENT; COMPUTER; SYSTEM;
SELECT; **PATH** ; **RELATED** ; OBJECT; BASED; ANALYSE; **RELATED** ; ASSOCIATE;
OBJECT

Derwent Class: T01; T06

International Patent Class (Main): G06F-007/00 ; G06F-009/00; G06F-017/30

International Patent Class (Additional): G06F-009/44

File Segment: EPI

14/5/14 (Item 14 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2004 Thomson Derwent. All rts. reserv.

013985117 **Image available**
WPI Acc No: 2001-469331/200151
XRPX Acc No: N01-348318

Object oriented program design/development assistance device
searches state transition path based on input start and completion
conditions, for generated state transition model

Patent Assignee: FUJITSU LTD (FUIT)
Number of Countries: 001 Number of Patents: 001
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
JP 2001166929	A	20010622	JP 99352824	A	19991213	200151 B

Priority Applications (No Type Date): JP 99352824 A 19991213

Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
JP 2001166929	A		12	G06F-009/06	

Abstract (Basic): JP 2001166929 A

NOVELTY - A state transition model generator (22) couples object state transition information on each object based on transmission and reception relationship of event for generating state transition model. A search unit (26) searches a suitable state transition path, based on input start condition corresponding to the model and completion condition.

DETAILED DESCRIPTION - An INDEPENDENT CLAIM is also included for memory medium storing test specification production program.

USE - Object oriented program design/development assistance device.

ADVANTAGE - Set -up conditions such as starting and completion conditions narrows down the test specification, thereby enables producing test specification, automatically without human error. Hence, reliability of testing is improved.

DESCRIPTION OF DRAWING(S) - The figure shows the explanatory drawing of program design/development assistance device. (Drawing includes non-English language text).

State transition model generator (22)

Search unit (26)

pp; 12 DwgNo 1/20

Title Terms: OBJECT; ORIENT; PROGRAM; DESIGN; DEVELOP; ASSIST; DEVICE;

SEARCH ; STATE; TRANSITION; PATH ; BASED; INPUT; START; COMPLETE;

CONDITION; GENERATE; STATE; TRANSITION; MODEL

Derwent Class: T01

International Patent Class (Main): G06F-009/06

International Patent Class (Additional): G06F-009/44; G06F-011/28

File Segment: EPI

14/5/17 (Item 17 from file: 350)
DIALOG(R) File 350:Derwent WPIX
(c) 2004 Thomson Derwent. All rts. reserv.

013167076 **Image available**

WPI Acc No: 2000-338949/200029

XRPX Acc No: N00-254473

Multi-dimensional object/relational database system for managing various types of business information, has search engine which retrieves data from fact table according to class property designated by user

Patent Assignee: ASPECT DEV (ASPE-N); I2 TECHNOLOGIES INC (ITWO-N); I2 TECHNOLOGIES US INC (ITWO-N)

Inventor: ALTHOFF J

Number of Countries: 090 Number of Patents: 006

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
WO 200019340	A1	20000406	WO 99US22674	A	19990930	200029 B
AU 200010978	A	20000417	AU 200010978	A	19990930	200035
EP 1125226	A1	20010822	EP 99954690	A	19990930	200149
			WO 99US22674	A	19990930	
KR 2001093775	A	20011029	KR 2001704144	A	20010330	200223
US 6366922	B1	20020402	US 98102463	A	19980930	200226
			US 99409069	A	19990930	
JP 2002526833	W	20020820	WO 99US22674	A	19990930	200258
			JP 2000572777	A	19990930	

Priority Applications (No Type Date): US 98102463 P 19980930; US 99409069 A 19990930

Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
WO 200019340	A1	E	25	G06F-017/30	
Designated States (National): AE AL AM AT AU AZ BA BB BG BR BY CA CH CN CR CU CZ DE DK DM EE ES FI GB GD GE GH GM HR HU ID IL IN IS JP KE KG KP KR KZ LC LK LR LS LT LU LV MD MG MK MN MW MX NO NZ PL PT RO RU SD SE SG SI SK SL TJ TM TR TT TZ UA UG UZ VN YU ZA ZW					
Designated States (Regional): AT BE CH CY DE DK EA ES FI FR GB GH GM GR IE IT KE LS LU MC MW NL OA PT SD SE SL SZ TZ UG ZW					
AU 200010978	A			G06F-017/30	Based on patent WO 200019340
EP 1125226	A1	E		G06F-017/30	Based on patent WO 200019340
Designated States (Regional): AL AT BE CH CY DE DK ES FI FR GB GR IE IT LI LT LU LV MC MK NL PT RO SE SI					
KR 2001093775	A			G06F-017/30	
US 6366922	B1			G06F-017/30	Provisional application US 98102463
JP 2002526833	W		26	G06F-017/30	Based on patent WO 200019340

Abstract (Basic): WO 200019340 A1

NOVELTY - A fact table (30) containing granular data (28) is represented along several dimensions to which **object - oriented classes** are associated. The **classes** map corresponding to data. A **search engine** **retrieves** data from table according to properties of the **classes** designated by user.

DETAILED DESCRIPTION - The **classes** include sub-class that **inherit** properties from **parent classes**. The **search engine** **retrieves** data corresponding to both unique and **inherited** property of sub-class, according to user's designation. A first panel of user interface illustrates the hierarchical relationship of **classes**. The properties of the selected class are displayed by second panel. An INDEPENDENT CLAIM is also included for user interface of multi-dimensional object/relational database system.

USE - For managing various types of business information.

ADVANTAGE - Enables user to create, edit and manipulate both application objects in database and application object relational model using suitable user interface. Due to **inheritance** properties of **classes**, users can construct **searches** based on properties associated with any of the **classes** and sub-**classes** within the hierarchy. The **search engine** includes information about structure of **classes** and tables hence user is enabled to relate data in terms of **classes** and

instances of **classes** rather than as rows in tables.

DESCRIPTION OF DRAWING(S) - The figure shows block diagram illustrating process of consolidation of data.

Granular data (28)

Fact table (30)

pp; 25 DwgNo 3/8

Title Terms: MULTI; DIMENSION; OBJECT; **RELATED** ; DATABASE; SYSTEM; MANAGE;
VARIOUS; TYPE; BUSINESS; INFORMATION; **SEARCH** ; ENGINE; **RETRIEVAL** ; DATA
; FACT; TABLE; ACCORD; CLASS; PROPERTIES; DESIGNATED; USER

Derwent Class: T01

International Patent Class (Main): G06F-017/30

International Patent Class (Additional): G06F-003/14; G06F-012/00

File Segment: EPI

14/5/22 (Item 22 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2004 Thomson Derwent. All rts. reserv.

012177531 **Image available**
WPI Acc No: 1998-594442/199850
Related WPI Acc No: 1999-070016
XRPX Acc No: N98-462584

Object - oriented queries processing method e.g. for object
relational database gateway - involves generating query for each super
class when number of super classes is greater than predetermined limit

Patent Assignee: INT BUSINESS MACHINES CORP (IBMC)

Inventor: PARVATHANENY B A; SRINIVASAN V

Number of Countries: 001 Number of Patents: 001

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
US 5829006	A	19981027	US 95465702	A	19950606	199850 B

Priority Applications (No Type Date): US 95465702 A 19950606

Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
US 5829006	A		35	G06F-017/30	

Abstract (Basic): US 5829006 A

The method involves receiving an **object oriented query** and prefetch **path** which identifies several target objects and prefetch **path** objects that are to be constructed, respectively. A **set** of relational **queries** is generated from the received **object oriented query**. When the generated **queries** are processed, tuples required to initialize base attributes of target objects are **retrieved** by RDBMS (12).

The target objects are defined by target class and by any super and sub **classes** of target class. A determining unit determines whether number of super **classes** is greater than predetermined limit and if so, **query** for each super class is generated. When generated **query** is processed, RDBMS is enabled to **retrieve** base attributes of each super class.

USE - For **retrieving** data from RDBMS.

ADVANTAGE - Conforms to class **inheritance** hierarchy defined by **object oriented** schemes.

Dwg.1/20

Title Terms: OBJECT; ORIENT; **QUERY** ; PROCESS; METHOD; OBJECT; **RELATED** ;
DATABASE; GATEWAY; GENERATE; **QUERY** ; SUPER; CLASS; NUMBER; SUPER; CLASS;
GREATER; PREDETERMINED; LIMIT

Derwent Class: T01

International Patent Class (Main): G06F-017/30

File Segment: EPI

14/5/23 (Item 23 from file: 350)
DIALOG(R) File 350:Derwent WPIX
(c) 2004 Thomson Derwent. All rts. reserv.

011931072 **Image available**
WPI Acc No: 1998-347982/199830
Related WPI Acc No: 1998-480747
XRPX Acc No: N98-271672

Object oriented query processing for RDBMS - involves translating
OO query to relational query and generating pre-fetch path

Patent Assignee: INT BUSINESS MACHINES CORP (IBMC)

Inventor: SRINIVASAN V

Number of Countries: 001 Number of Patents: 001

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
US 5765159	A	19980609	US 94366238	A	19941229	199830 B
			US 97870581	A	19970521	

Priority Applications (No Type Date): US 94366238 A 19941229; US 97870581 A
19970521

Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
US 5765159	A		32	G06F-017/30	Cont of application US 94366238

Abstract (Basic): US 5765159 A

The query processing method involves receiving an object - oriented query and at least one prefetch path from an object - oriented source, the at least one prefetch path identifying one or more objects which are desired to be constructed. The object - oriented query is translated to a translated object query, the translated object query being a relational database query capable of retrieving from the RDBMS data to initialize base attributes of top-level objects identified by the object - oriented query.

Generating from the translated object query and the at least one prefetch path a set of relational queries capable of retrieving from the RDBMS data to construct the objects identified by the at least one prefetch path. The RDBMS processes the set of relational queries.

ADVANTAGE - Allows object oriented data access.

Dwg. 7/13

Title Terms: OBJECT; ORIENT; QUERY; PROCESS; TRANSLATION; QUERY;
RELATED; QUERY; GENERATE; PRE; FETCH; PATH

Derwent Class: T01

International Patent Class (Main): G06F-017/30

File Segment: EPI

14/5/24 (Item 24 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2004 Thomson Derwent. All rts. reserv.

011223888 **Image available**
WPI Acc No: 1997-201813/199718
Related WPI Acc No: 1995-382714
XRPX Acc No: N97-166881

Object - oriented computer architecture using directory objects - uses
objects including directed graph of directory objects for locating
objects and contg. object names and object pointers for locating other
objects in memory, and root directory object

Patent Assignee: DIGITAL EQUIP CORP (DIGI)

Inventor: JENNESS S M

Number of Countries: 001 Number of Patents: 001

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
US 5615363	A	19970325	US 9384292	A	19930628	199718. B
			US 95456711	A	19950601	

Priority Applications (No Type Date): US 9384292 A 19930628; US 95456711 A
19950601

Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
US 5615363	A	11	G06F-017/30		Cont of application US 9384292 Cont of patent US 5463774

Abstract (Basic): US 5615363 A

The objects stored in computer's memory include a directed graph of
object directories. Each object directory stores object names and
object pointers for locating and accessing other objects. A root
directory object, which is the starting point for locating any
specified object, stores object names and object pointers to a set of
first level object directories. Each object has an associated path
-name that defines a path through the directed graph of object
directories for accessing that object.

Each path -name is a succession of path elements, proceeding
from a first path element to a last path element. A default path
-name parsing procedure is used for parsing any specified object's
path -name, starting with the first path element, until the default
path -name parsing procedure accesses an object directory having its
own distinct path -name parsing procedure. At least one object
directory has its own distinct path -name parsing procedure for
locating objects whose path -name includes a path element that
identifies that object directory. When parsing a specified object path
-name, the path elements are successively parsed; one at a time by
accessing the corresponding object directory, and then using the
information in that object directory to access the object or directory
object associated with a next path element. When an object directory
is accessed that has its own path -name parsing procedure, that object
directory's path -name parsing procedure is invoked to parse the
remaining path elements of the specified object path -name.

Dwg.2/7

Title Terms: OBJECT; ORIENT; COMPUTER; ARCHITECTURE; DIRECTORY; OBJECT;
OBJECT; DIRECT; GRAPH; DIRECTORY; OBJECT; LOCATE ; OBJECT; CONTAIN;
OBJECT; NAME; OBJECT; POINT; LOCATE ; OBJECT; MEMORY; ROOT; DIRECTORY;
OBJECT

Derwent Class: T01

International Patent Class (Main): G06F-017/30

File Segment: EPI

14/5/26 (Item 26 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2004 Thomson Derwent. All rts. reserv.

010481393 **Image available**
WPI Acc No: 1995-382714/199549
Related WPI Acc No: 1997-201813
XRPX Acc No: N95-280367

Locating object in object - oriented computer architecture using
directory objects - parsing path elements in specified object path
-name by accessing identified directory objects, object using last parsed
path element and determining when directory objects include path -name
procedure and invoking for objects not parsed

Patent Assignee: DIGITAL EQUIP CORP (DIGI)

Inventor: JENNESS S M

Number of Countries: 001 Number of Patents: 001

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
US 5463774	A	19951031	US 9384292	A	19930628	199549 B

Priority Applications (No Type Date): US 9384292 A 19930628

Patent Details:

Patent No	Kind	Lan Pg	Main IPC	Filing Notes
US 5463774	A	12	G06F-017/30	

Abstract (Basic): US 5463774 A

The method involves storing objects comprising data structures in memory. A directed graph of the directory objects is also stored in the memory for locating and accessing these objects. Each directory object stores object names and object pointers for locating and accessing other objects stored in the memory. The directory objects include a root directory object, which includes object names and object pointers to a set of first level directory objects.

Each object stored in the memory has an associated path -name that defines a path through the directed graph of the directory objects for accessing each object. Each path -name comprises a succession of path elements that includes first and last path elements.

Each directory object is provided with a distinct path -name parsing procedure for locating those of the multiplicity of objects whose path -name includes a path element, other than the last path element, identifying each directory object. The computer system receives a specified object path -name and successively parses the path elements in the specified object path -name by accessing the successive directory objects identified by the successively parsed path elements, accessing the object identified by the specified object path -name when the last path element is parsed, and determining when any of the accessed directory objects includes a distinct path -name parsing procedure and invoking the distinct path -name parsing procedure of the accessed directory object to parse the path elements of the specified object path -name, which is not yet parsed.

ADVANTAGE - Allows usage of variety of path -name conventions and object location procedures that differ from the default convention and procedure e.g. as process objects and thread objects are extremely ephemeral, data structures and procedures for locating objects are best implemented differently than data structures and procedures used to locate other types of objects. Also used by different filing subsystems for locating file objects.

Dwg.7/7

Title Terms: LOCATE ; OBJECT; OBJECT; ORIENT; COMPUTER; ARCHITECTURE;
DIRECTORY; OBJECT; PARSE; PATH ; ELEMENT; SPECIFIED; OBJECT; PATH ;
NAME; ACCESS; IDENTIFY; DIRECTORY; OBJECT; OBJECT; LAST; PATH ; ELEMENT;
DETERMINE; DIRECTORY; OBJECT; PATH ; NAME; PROCEDURE; INVOKE; OBJECT

Derwent Class: T01

International Patent Class (Main): G06F-017/30

File Segment: EPI

14/5/28 (Item 28 from file: 350)
DIALOG(R)File 350:Derwent WPIX
(c) 2004 Thomson Derwent. All rts. reserv.

009879577 **Image available**

WPI Acc No: 1994-159491/199419

XRPX Acc No: N94-125394

Class inheritance solution process for object - oriented data processing and programming system - referring to status of target superclass in storage to determine if search for method is to be continued using search target superclass having lower priority

Patent Assignee: HITACHI LTD (HITA); HITACHI MICROCOMPUTER SYSTEM (HITA-N)

Inventor: MOKI K; NAMIOKA M; SATOH K; YAMAMOTO Y

Number of Countries: 001 Number of Patents: 001

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
US 5313630	A	19940517	US 91690592	A	19910423	199419 B

Priority Applications (No Type Date): JP 90107601 A 19900425

Patent Details:

Patent No	Kind	Lan Pg	Main IPC	Filing Notes
US 5313630	A	17	G06F-015/40	

Abstract (Basic): US 5313630 A

The **object - oriented** database management system is connected to a number of databases and includes a class definition unit, a static **inheritance** processing unit within the class definition unit, a message processing unit, a dynamic **inheritance** processing unit within the message processing unit, and an object management unit. The **object - oriented** database management system defines the **inheritance** relationship between classes and an arbitrary class and the **inheritance** priority order.

The class is a basic unit of programming, and executes the data processing by solving the **inheritance** relationship between classes in accordance with the priority order. For each class is prepared updatable **inheritance** solution status information which the static **inheritance** processing unit causes to be stored in a table. The **inheritance** solution status information includes status information and time information. The status information is representative of whether or not **inheritance** between the class and the superclasses can be completely solved, and the time information is representative of a time when the status is established.

ADVANTAGE - Both partial **inheritance** processing and **inheritance** re-processing to be executed as development progresses can be reliably and efficiently performed in accordance with priority order of superclasses.

Dwg.3/9

Title Terms: CLASS; SOLUTION; PROCESS; OBJECT; ORIENT; DATA; PROCESS; PROGRAM; SYSTEM; REFER; STATUS; TARGET; STORAGE; DETERMINE; **SEARCH** ; METHOD; CONTINUE; **SEARCH** ; TARGET; LOWER; PRIORITY

Derwent Class: T01

International Patent Class (Main): G06F-015/40

International Patent Class (Additional): **G06F-007/00**

File Segment: EPI

Set	Items	Description
S1	131966	OO OR OBJECT()ORIENT? OR OODB OR C()PLUS OR CPLUS OR ORDBMS OR ORDB
S2	7548315	PATH? OR ROUTE? OR RELATED OR INHERIT? OR CHILD? OR PARENT? OR ANCESTOR? OR INSTANCE(2N)LOCATION?
S3	3133980	QUER? OR SEARCH? OR SEEK? OR FIND? OR LOCATE OR LOCATEES OR LOCATING OR LOCATE OR RETRIEV?
S4	2163125	SETS OR SET OR BAGS OR CLASSES
S5	902553	OBJECT OR OBJECTS
S6	863	S2(3N)S3(3N)S5
S7	97	S1 AND S4 AND S6
S8	67	RD (unique items)
S9	63	S8 NOT PY>2000
S10	63	S9 NOT PD=20001012:20021012
S11	63	S10 NOT PD=20021012:20051011
S12	27219	S2(N) (S3 OR S5)
S13	14	S11 AND S12
S14	64538	S2(2N) (S3 OR S5)
S15	47	S14 AND S11
File	8: Ei Compendex(R) 1970-2004/May W1	(c) 2004 Elsevier Eng. Info. Inc.
File	35: Dissertation Abs Online 1861-2004/Apr	(c) 2004 ProQuest Info&Learning
File	65: Inside Conferences 1993-2004/May W2	(c) 2004 BLDSC all rts. reserv.
File	2: INSPEC 1969-2004/May W1	(c) 2004 Institution of Electrical Engineers
File	94: JICST-EPlus 1985-2004/Apr W3	(c) 2004 Japan Science and Tech Corp(JST)
File	111: TGG Natl. Newspaper Index(SM) 1979-2004/May 12	(c) 2004 The Gale Group
File	233: Internet & Personal Comp. Abs. 1981-2003/Sep	(c) 2003 EBSCO Pub.
File	6: NTIS 1964-2004/May W2	(c) 2004 NTIS, Intl Cpyrght All Rights Res
File	144: Pascal 1973-2004/May W1	(c) 2004 INIST/CNRS
File	34: SciSearch(R) Cited Ref Sci 1990-2004/May W1	(c) 2004 Inst for Sci Info
File	62: SPIN(R) 1975-2004/Mar W3	(c) 2004 American Institute of Physics
File	99: Wilson Appl. Sci & Tech Abs 1983-2004/Apr	(c) 2004 The HW Wilson Co.

15/5/2 (Item 2 from file: 8)
DIALOG(R) File 8: Ei Compendex(R)
(c) 2004 Elsevier Eng. Info. Inc. All rts. reserv.

05642682 E.I. No: EIP00095307008

Title: **VOQL: A Visual Object - oriented database Query Language for visualizing path expressions**

Author: Kim, Jeonghee; Han, Taisook; Lee, Suk Kyoan

Corporate Source: Korea Advanced Inst of Science and Technology, Taejon, South Korea

Source: Computer Systems Science and Engineering v 15 n 4 Jul 2000. p 215-232

Publication Year: 2000

CODEN: CSSEI ISSN: 0267-6192

Language: English

Document Type: JA; (Journal Article) Treatment: T; (Theoretical)

Journal Announcement: 0010W2

Abstract: Expressing complex query expressions in a concise and intuitive way has been a challenge in the design of **object - oriented** query languages. Since class hierarchies are usually represented as graphs, many graph-based visual query languages have been proposed. However, partly due to lack of well-defined syntax and semantics of languages and partly due to improper representation of their visual constructs, those approaches fail to meet the challenge. In this paper, we propose a graph-based, visual **object - oriented** query language called Visual Object Query Language (VOQL). VOQL has such good features as simple and intuitive syntax, well-defined semantics, and excellent expressive power for **sets**, which few visual languages have. Basic visual constructs of the language are blobs, subblobs, binding edges, and flattening edges. Blobs and subblobs are used to denote **sets** of **objects** that **path** expressions represent. Binding edges and flattening edges are designed to visually simulate the notion of variable binding and the notion of dot functions in path expressions. Based on the basic visual constructs, VOQL terms, VOQL formulas, and VOQL expressions are defined in a way as in tuple relational calculus. Their semantics are provided by syntax directed translation to corresponding counterparts in **Object - Oriented** Predicate Calculus (OOPC). (Author abstract) 22 Refs.

Descriptors: Relational database systems; Query languages; **Object oriented** programming; Computer systems programming; Graph theory; Computational linguistics

Identifiers: Visual object query languages (VOQL)

Classification Codes:

723.3 (Database Systems); 723.1 (Computer Programming); 921.4 (Combinatorial Mathematics, Includes Graph Theory, Set Theory); 721.1 (Computer Theory, Includes Formal Logic, Automata Theory, Switching Theory, Programming Theory)

723 (Computer Software); 921 (Applied Mathematics); 721 (Computer Circuits & Logic Elements)

72 (COMPUTERS & DATA PROCESSING); 92 (ENGINEERING MATHEMATICS)

15/5/3 (Item 3 from file: 8)
DIALOG(R)File 8: Ei Compendex(R)
(c) 2004 Elsevier Eng. Info. Inc. All rts. reserv.

05351841 E.I. No: EIP99094775158

Title: Performance analysis of parallelization models for path expression queries

Author: Taniar, David; Rahayu, J. Wenny

Corporate Source: RMIT Univ, Melbourne, Aust

Source: Information Sciences v 117 n 1 1999. p 107-142

Publication Year: 1999

CODEN: ISIJBC ISSN: 0020-0255

Language: English

Document Type: JA; (Journal Article) Treatment: T; (Theoretical)

Journal Announcement: 9910W3

Abstract: In this paper, parallelization models for path expressions queries are studied. Path expression queries involve multiple classes along aggregation/association hierarchies. Parallelization models for path expression queries are 'inter-object parallelization' and 'inter-class parallelization'. Inter-object parallelization exploits the associativity within complex objects, whereas inter-class parallelization imposes upon process independence. The behaviours of these parallelization models are described in terms of analytical models. Performance evaluation is also performed to confirm the results from the quantitative analysis. (Author abstract) 22 Refs.

Descriptors: Parallel processing systems; Mathematical models; Database systems; Object oriented programming; Performance

Identifiers: Path expression queries; Inter object parallelization; Inter class parallelization; Parallel query processing

Classification Codes:

722.4 (Digital Computers & Systems); 921.6 (Numerical Methods); 723.3 (Database Systems); 723.1 (Computer Programming)

722 (Computer Hardware); 921 (Applied Mathematics); 723 (Computer Software)

72 (COMPUTERS & DATA PROCESSING); 92 (ENGINEERING MATHEMATICS)

15/5/4 (Item 4 from file: 8)
DIALOG(R)File 8:EI Compendex(R)
(c) 2004 Elsevier Eng: Info. Inc. All rts. reserv.

04820823 E.I. No: EIP97093824188

Title: Query modification in object - oriented database federations

Author: Vermeer, Mark W.W.; Apers, Peter M.G.

Corporate Source: Univ of Twente, Enschede, Neth

Conference Title: Proceedings of the 1997 2nd IFCIS International
Conference on Cooperative Information Systems, CoopIS 97

Conference Location: Kiawah Island, SC, USA Conference Date:
19970624-19970627

Sponsor: IEEE

E.I. Conference No.: 46968

Source: Proceedings of the IFCIS International Conference on Cooperative
Information Systems, CoopIS 1997. IEEE, Los Alamitos, CA, USA, 97TB100143. p
193-202

Publication Year: 1997

CODEN: 002663

Language: English

Document Type: CA; (Conference Article) Treatment: G; (General Review);
T; (Theoretical)

Journal Announcement: 9711W1

Abstract: We discuss the modification of queries against an integrated
view in a federation of **object - oriented** databases. We present a
generalization of existing algorithms for simple global query processing
that works for arbitrarily defined integration **classes**. We then extend
this algorithm to deal with **object - oriented** features such as **queries**
involving **path** expressions and nesting. We show how properties of the OO
-style of modelling relationships through object references can be
exploited to reduce the number of subqueries necessary to evaluate such
queries. (Author abstract) 19 Refs.

Descriptors: Distributed database systems; Information retrieval; **Object
oriented** programming; Algorithms; Query languages; Computer simulation

Identifiers: **Object oriented** databases

Classification Codes:

723.3 (Database Systems); 903.3 (Information Retrieval & Use); 723.1
(Computer Programming); 723.5 (Computer Applications)

723 (Computer Software); 903 (Information Science)

72 (COMPUTERS & DATA PROCESSING); 90 (GENERAL ENGINEERING)

15/5/6 (Item 6 from file: 8)
DIALOG(R)File 8:Ei Compendex(R)
(c) 2004 Elsevier Eng. Info. Inc. All rts. reserv.

04426390 E.I. No: EIP96063215658

Title: Enhanced nested-inherited index for OODBMS
Author: Bertino, E.; Salerno, S.; Shidlovsky, B.
Corporate Source: Universita di Milano, Italy
Conference Title: Proceedings of the 1995 ACM CIKM 4th International Conference on Information and Knowledge Management
Conference Location: Baltimore, MD, USA **Conference Date:** 19951128-19951202
Sponsor: ACM; SIGIR; SIGART
E.I. Conference No.: 44794
Source: International Conference on Information and Knowledge Management, Proceedings 1995. ACM, New York, NY, USA. p 58-65
Publication Year: 1995
CODEN: 002176
Language: English
Document Type: CA; (Conference Article) **Treatment:** A; (Applications); T; (Theoretical)
Journal Announcement: 9608W2

Abstract: The nested-inherited index has been recently proposed as an access structure providing an integrated support for **queries** in **object - oriented** databases along both aggregation and **inheritance** hierarchies. It is very efficient for retrieval operations. However, its high update costs make this structure suitable only for hierarchies with a small number of **classes**. In this paper we propose an enhanced nested-inherited index, able to support update operations more efficiently, whereas supporting nested predicates as efficiently as the nested-inherited index. The new organization supports the construction of several index allocation strategies, from which the most efficient with respect to a given workload can be selected. The new and old indices are compared using an analytical cost model. Results of the analysis show that the enhanced nested-inherited index provides superior performance than the inherited-multiindex and nested-inherited index. (Author abstract) 11 Refs.

Descriptors: Data structures; Database systems; **Object oriented** programming; Query languages; Information retrieval; Indexing (of information); Mathematical models; Hierarchical systems; Performance
Identifiers: Nested **inherited** index; **Object oriented** programming systems; Aggregation hierarchies; Inheritance hierarchies; Workload; Insertion time

Classification Codes:

723.2 (Data Processing); 723.3 (Database Systems); 723.1 (Computer Programming); 903.1 (Information Sources & Analysis); 921.6 (Numerical Methods)

723 (Computer Software); 903 (Information Science); 921 (Applied Mathematics)

72 (COMPUTERS & DATA PROCESSING); 90 (GENERAL ENGINEERING); 92 (ENGINEERING MATHEMATICS)

15/5/8 (Item 8 from file: 8)
DIALOG(R) File 8: Ei Compendex(R)
(c) 2004 Elsevier Eng. Info. Inc. All rts. reserv.

03436171 E.I. Monthly No: EI9206077845

Title: Parallel Petri net path searching on a local area network.

Object oriented Pascal implementation.

Author: Anneberg, L.; Rajappan, G.; Singh, H.; Yaprak, E.

Corporate Source: Wayne State Univ, Detroit, MI, USA

Source: Microelectronics and Reliability v 32 n 1-2 Jan-Feb 1992 p 199-206

Publication Year: 1992

CODEN: MCRLAS **ISSN:** 0026-2714

Language: English

Document Type: JA; (Journal Article) **Treatment:** T; (Theoretical); A; (Applications)

Journal Announcement: 9206

Abstract: In this increasingly technological age, the need for parallel processing has never been greater than it is today. Parallelizing a particular program or system can increase the speed of the system by the number of processors (in the ideal case). A wide variety of parallel and distributed systems exist, and their efficiency may not be the same for different types of problems. In fact, the variation of real world systems and their possible implementation on an actual distributed system is enormous. In this paper, an inexpensive solution to parallelizing the determination of paths in a Petri net is presented. The system utilized is an IBM Local Area Network with four computers arranged as a parallel processing system. A three step generalized algorithm to accomplish the parallelization is given. The actual implementation is facilitated by the use of **object oriented** software design, in particular Turbo Pascal version 5.5 (which supports the creation of objects and their associated data **sets** and procedures). Finally, actual implementation and object creation of this system is accomplished. (Author abstract) 8 Refs.

Descriptors: *MATHEMATICAL TECHNIQUES--*Petri Nets; COMPUTER NETWORKS--Local Area Networks; COMPUTER SYSTEMS, DIGITAL--Parallel Processing; COMPUTER PROGRAMMING LANGUAGES--PASCAL

Identifiers: **OBJECT ORIENTED SOFTWARE;** TURBO PASCAL VERSION 5.5

Classification Codes:

921 (Applied Mathematics); 722 (Computer Hardware); 723 (Computer Software); 716 (Radar, Radio & TV Electronic Equipment); 717 (Electro-Optical Communications); 718 (Telephone & Line Communications)
92 (ENGINEERING MATHEMATICS); 72 (COMPUTERS & DATA PROCESSING); 71 (ELECTRONICS & COMMUNICATIONS)

15/5/15 (Item 5 from File: 35)
DIALOG(R)File 35:Dissertation Abs Online
(c) 2004 ProQuest Info&Learning. All rts. reserv.

01328428 ORDER NO: AAD94-01728

ALGORITHMS FOR GENERATION OF PATH -METHODS IN OBJECT - ORIENTED DATABASES

Author: MEHTA, ASHISH KHANDUBHAI

Degree: PH.D.

Year: 1993

Corporate Source/Institution: NEW JERSEY INSTITUTE OF TECHNOLOGY (0152)

ADVISERS: YEHOShUA PERL; JAMES GELLER

Source: VOLUME 54/08-B OF DISSERTATION ABSTRACTS INTERNATIONAL.

PAGE 4254. 226 PAGES

Descriptors: COMPUTER SCIENCE

Descriptor Codes: 0984

A **path -method** is a mechanism in **object - oriented** databases (OODBs) to **retrieve** or to update information relevant to one class that is not stored with that class but with some other class. A path-method is a method which traverses from one class through a chain of connections between **classes** to access information at another class. However, it is a difficult task for a user to write path-methods, because it might require comprehensive knowledge of many **classes** of the conceptual schema, while a typical user has often incomplete or even inconsistent knowledge of the schema.

This dissertation proposes an approach to the generation of path-methods in an **OODB** to solve this problem. We have developed the Path-Method Generator (PMG) system, which generates path-methods according to a naive user's requests. PMG is based on access weights which reflect the relative frequency of the connections and precomputed access relevance between every pair of **classes** of the **OODB** computed from access weights of the connections. We present specific rules for access weight assignment, efficient algorithms to compute access relevance in a single **OODB**, and a variety of traversal algorithms based on access weights and precomputed access relevance. Experiments with a university environment **OODB** and a sample of path-methods identify some of these algorithms as very successful in generating most of the desired path-methods. Thus, the PMG system is an efficient tool for aiding the user with the difficult task of querying and updating a large **OODB**.

The path-method generation in an interoperable multi **object - oriented** database (IM- **OODB**) is even more difficult than for a single **OODB**, since a user has to be familiar with several OODBs. We use a hierarchical approach for deriving efficient online algorithms for the computation of access relevance in an IM- **OODB**, based on precomputed access relevance for each autonomous **OODB**. In an IM- **OODB** the access relevance is used as guide in generating path-methods between the **classes** of different OODBs.

15/5/17 (Item 1 from file: 65)
DIALOG(R)File 65:Inside Conferences
(c) 2004 BLDSC all rts. reserv. All rts. reserv.

03918104 INSIDE CONFERENCE ITEM ID: CN041163410

**Parallelization Models for Path Expression Queries in Object -
Oriented Databases**

Taniar, D.

CONFERENCE: Vol 1; Fuzzy logic, intelligent control & genetic
algorithm; Vol 2; Computational intelligence, neural networks &
semiotics; Vol 3; Rough set & computer science Joint conference
of information sciences

P: 215-218

Association for Intelligent Machinery, 1997

ISBN: 0964345668

LANGUAGE: English DOCUMENT TYPE: Conference Papers

CONFERENCE LOCATION: Research Triangle Park, NC 1997; Mar (199703) (
199703)

BRITISH LIBRARY ITEM LOCATION: m01/36499 = vol 3

NOTE:

Also known as JCIS '97

DESCRIPTORS: JCIS; information sciences; fuzzy logic; computational
intelligence; rough **set**

15/5/19 (Item 2 from File: 2)

DIALOG(R)File 2:INSPEC

(c) 2004 Institution of Electrical Engineers. All rts. reserv.

6912180 INSPEC Abstract Number: C2001-06-6160J-002

Title: Indexing inheritance and aggregation

Author(s): Kang, U.T.; Davis, K.C.; Ravishankar, S.

Author Affiliation: Database Syst. Lab., Cincinnati Univ., OH, USA

Conference Title: Proceedings of the Ninth International Conference on Information and Knowledge Management. CIKM 2000 p.110-18

Editor(s): Agah, A.; Callan, J.; Rundensteiner, E.

Publisher: ACM, New York, NY, USA

Publication Date: 2000 Country of Publication: USA xvi+532 pp.

ISBN: 1 58113 320 0 Material Identity Number: XX-2000-03146

U.S. Copyright Clearance Center Code: 1 58113 320 0/2000/0011...\$5.00

Conference Title: Proceedings of Ninth International Conference on Information and Knowledge Management (CIKM)

Conference Sponsor: ACM

Conference Date: 6-11 Nov. 2000 Conference Location: McLean, VA, USA

Language: English Document Type: Conference Paper (PA)

Treatment: Practical (P)

Abstract: We propose an index called the path inherited dictionary index (PIDI) that supports **queries** along both aggregation (**paths**) and **inheritance** for **retrieving inherited nested objects** . A structural description of PIDI and retrieval examples are presented here. To evaluate the effectiveness of PIDI we present a **set** of integrated query types and a cost model for storage and retrieval. The query types consider combinations of (1) forward, reverse, and mixed traversal, (2) traversal of some or all of an inheritance hierarchy, (3) partial **path retrieval** , (4) **queries** over nested attributes or any OID along a path, and (5) range and point queries. The focus of the paper is on the new data structure itself and how to navigate it to answer different kinds of integrated queries. We give an overview of a performance study comparing PIDI to the nested-inheritance index (NIX). We conclude that PIDI performs well on a variety of integrated query types, although in many cases occupies more space than NIX. (11 Refs)

Subfile: C

Descriptors: database indexing; inheritance; **object - oriented** databases

Identifiers: path inherited dictionary index; **inherited nested objects** ; structural description; integrated query types; cost model; query types; forward; reverse; mixed traversal; inheritance hierarchy; partial **path retrieval** ; nested attributes; nested-inheritance index

Class Codes: C6160J (Object-oriented databases); C6120 (File organisation); C6110J (Object-oriented programming); C4250 (Database theory)

Copyright 2001, IEE

15/5/24 (Item 7 from File: 2)
DIALOG(R) File 2:INSPEC
(c) 2004 Institution of Electrical Engineers. All rts. reserv.

6240821 INSPEC Abstract Number: C1999-06-6140D-017

Title: Visualization of path expressions in a visual object - oriented database query language

Author(s): Jeonghee Kim; Taisook Han; Suk Kyoon Lee

Author Affiliation: Dept. of Comput. Sci., Korea Adv. Inst. of Sci. & Technol., Seoul, South Korea

Conference Title: Proceedings. 6th International Conference on Advanced Systems for Advanced Applications p.99-108

Editor(s): Chen, A.L.P.; Lochovsky, F.H.

Publisher: IEEE Comput. Soc, Los Alamitos, CA, USA

Publication Date: 1999 Country of Publication: USA xii+356 pp.

ISBN: 0 7695 0084 6 Material Identity Number: XX-1999-01043

U.S. Copyright Clearance Center Code: 0 7695 0084 6/99/\$10.00

Conference Title: Proceedings. 6th International Conference on Database Systems for Advanced Applications

Conference Sponsor: Nat. Tsing Hua Univ.; Nat. Sci. Council; Minstr. Educ.; Inf. Process. Soc. Japan

Conference Date: 19-21 April 1999 Conference Location: Hsinchu, Taiwan

Language: English Document Type: Conference Paper (PA)

Treatment: Theoretical (T)

Abstract: Path expressions have been accepted for concisely manipulating the nested structures in complex **object - oriented** query expressions. However, previous visual query languages hardly represent such query expressions in a concise and intuitive way partly due to improper visual representation of path expressions and partly due to lack of well-defined system and semantics of languages. In this paper, we present visual modeling of **path** expressions in a visual **object - oriented** database **query** language called Visual **Object - Oriented** Query Language (VOQL) which has excellent expressive power for **sets**, simple and intuitive syntax, and well-defined semantics. This is enabled by explicitly specifying the semantics of multi-valued path expressions based on the visual notation capable of representing **set** relationships in addition to functional relationships. The basic visual constructs called blobs and nested blobs denote **sets** of **objects** that **path** expressions represent while the constructs called binding edges and flattening edges visually simulate the notions of variable binding and dot functions in path expressions respectively. Based on the constructs, the grammar of VOQL defines the syntactic components while the semantics of query expressions are provided by syntax-directed translation to the counterparts in the extended relational calculus. Also, the visual constructs allow modeling of restricted universal quantification with a visual scoping box and effectively represent nested quantification and recursive queries without semantic ambiguities. (26 Refs)

Subfile: C

Descriptors: data visualisation; grammars; **object - oriented** languages; programming language semantics; query languages; visual languages

Identifiers: path expression visualization; visual **object - oriented** database query language; nested structures; complex **object - oriented** query expressions; semantics; visual modeling; Visual **Object - Oriented** Query Language; intuitive syntax; well-defined semantics; multi-valued path expressions; **set** relationships; blobs; nested blobs; binding edges; flattening edges; visual simulation; variable binding functions; variable dot functions; grammar; syntax-directed translation; extended relational calculus; restricted universal quantification; visual scoping box; recursive queries

Class Codes: C6140D (High level languages); C4210L (Formal languages and computational linguistics); C6130B (Graphics techniques); C6160 (Database management systems (DBMS)); C6110J (Object-oriented programming); C6110V (Visual programming)

Copyright 1999, IEE

15/5/25 (Item 8 from File: 2)

DIALOG(R) File 2:INSPEC

(c) 2004 Institution of Electrical Engineers. All rts. reserv.

6110876 INSPEC Abstract Number: C9901-6160J-023

Title: Query **optimization techniques utilizing path indexes in object - oriented database systems**

Author(s): Wan-Sup Cho; Kyu-Young Whang; Seung-Sun Lee; Yong-Ik Yoon

Author Affiliation: Real-Time OS Sect., ETRI, Taejon, South Korea

Conference Title: Database Systems for Advanced Applications '97. Proceedings of the Fifth International Conference p.21-9

Editor(s): Topor, R.; Tanaka, K.

Publisher: World Scientific, Singapore

Publication Date: 1997 Country of Publication: Singapore xviii+542 pp.

ISBN: 981 02 3107 5 Material Identity Number: XX97-01303

Conference Title: Proceedings of 5th International Conference on Database Systems for Advanced Applications

Conference Sponsor: Royal Melbourne Inst. Technol.; Univ. Melbourne; Inf. Process. Soc. Japan; CSIRO; Ferntree Comput. Corp

Conference Date: 1-4 April 1997 Conference Location: Melbourne, Vic., Australia

Language: English Document Type: Conference Paper (PA)

Treatment: Practical (P); Theoretical (T)

Abstract: We propose query optimization techniques that fully utilize the advantages of **path indexes in object - oriented** database systems. Although path indexes provide an efficient access to complex objects, little research has been done on query optimization that fully utilize path indexes. We first devise a generalized index intersection technique, adapted to the structure of the path index extended from conventional indexes, for utilizing multiple (path) indexes to access each class in a query. We then propose the query graph reduction algorithm that replaces the **classes** in the **query** graph with **path** index scans; we call the resultant query graph reduced query graph (RQG). We finally present the search algorithm that finds the least-cost evaluation plan from RQG based on a cost model. The proposed query optimization techniques enhance database performance by using path indexes instead of direct accesses to data in the evaluating queries. (24 Refs)

Subfile: C

Descriptors: database indexing; database theory; graph theory; **object - oriented** databases; optimisation; query processing

Identifiers: query optimization; path indexes; **object - oriented** database; complex object access; generalized index intersection; multiple indexes; query graph reduction algorithm; reduced query graph; search algorithm; least-cost evaluation plan; database performance

Class Codes: C6160J (Object-oriented databases); C4250 (Database theory)

Copyright 1998, IEE

L Number	Hits	Search Text	DB	Time stamp
1	126	object\$1 near3 names same oriented	USPAT; US-PGPUB; JPO	2004/05/12 23:40
2	9	707/103R.ccls. and (object\$1 near3 names same oriented)	USPAT; US-PGPUB; JPO	2004/05/12 23:40
3	5	707/3.ccls. and (object\$1 near3 names same oriented)	USPAT; US-PGPUB; JPO	2004/05/12 23:41
4	6	707/100.ccls. and (object\$1 near3 names same oriented)	USPAT; US-PGPUB; JPO	2004/05/12 23:41
5	8	707/102.ccls. and (object\$1 near3 names same oriented)	USPAT; US-PGPUB; JPO	2004/05/12 23:41
-	1	("20020046208").PN.	USPAT; US-PGPUB; JPO	2004/05/05 10:36
-	0	("objectssameorientedssamerelationshipsamequerysamepathsameanal\$5").PN.	USPAT; US-PGPUB; JPO	2004/05/05 10:40
-	55932	objects same oriented same relationship same query same relateds ame path same anal\$5	USPAT; US-PGPUB; JPO	2004/05/11 20:31
-	0	objects same oriented same relationship same query same related same path same anal\$5	USPAT; US-PGPUB; JPO	2004/05/05 10:40
-	0	objects same oriented same relationship same query same related same path	USPAT; US-PGPUB; JPO	2004/05/11 22:15
-	1	objects same oriented same relationship same query same related	USPAT; US-PGPUB; JPO	2004/05/05 12:55
-	2	objects same oriented same relationship same query same path	USPAT; US-PGPUB; JPO	2004/05/08 13:59
-	24	objects same oriented same relationship same query	USPAT; US-PGPUB; JPO	2004/05/12 10:44
-	7	("6208993" or "5787275" or "5809505" or "6182277" or "5893108").pn. or "2002010840"	USPAT; US-PGPUB; JPO	2004/05/05 13:21
-	6	("6208993" or "5787275" or "5809505" or "6182277" or "5893108").pn. or "20020107840"	USPAT; US-PGPUB; JPO	2004/05/06 07:54
-	7	("6208993" or "5787275" or "5809505" or "6182277" or "5893108").pn. or "2002010840"	USPAT; US-PGPUB; JPO	2004/05/06 07:58
-	0	((("6208993" or "5787275" or "5809505" or "6182277" or "5893108").pn. or "2002010840") and (object same oriented same relationships same query same path)	USPAT; US-PGPUB; JPO	2004/05/06 07:55
-	3	object same oriented same relationships same query same path	USPAT; US-PGPUB; JPO	2004/05/06 07:58
-	0	((("6208993" or "5787275" or "5809505" or "6182277" or "5893108").pn. or "2002010840") and (object same relationships same query same path)	USPAT; US-PGPUB; JPO	2004/05/06 07:58
-	2	((("6208993" or "5787275" or "5809505" or "6182277" or "5893108").pn. or "2002010840") and (object same query same path)	USPAT; US-PGPUB; JPO	2004/05/08 13:22
-	3	cost same path same object same select	USPAT; US-PGPUB; JPO	2004/05/08 13:22
-	0	cost same path same object same select same relationship	USPAT; US-PGPUB; JPO	2004/05/08 13:22

09/973195

-	0	(lowest near2 cost) same select same (path or directory) same object	USPAT; US-PGPUB; JPO	2004/05/08 13:27
-	76	(lowest near2 cost) same select same (path or directory)	USPAT; US-PGPUB; JPO	2004/05/08 13:27
-	1	(lowest near2 cost) same select same (path or directory) same relation\$	USPAT; US-PGPUB; JPO	2004/05/08 13:28
-	31	query same string same name same items	USPAT; US-PGPUB; JPO	2004/05/08 14:00
-	1	query same string same name same items same objects	USPAT; US-PGPUB; JPO	2004/05/08 14:01
-	10	query same string same name same items same database	USPAT; US-PGPUB; JPO	2004/05/08 14:03
-	0	"a string of name items" same query	USPAT; US-PGPUB; JPO	2004/05/08 14:04
-	0	"a string of name" same query	USPAT; US-PGPUB; JPO	2004/05/08 14:04
-	31	string same name same items same query	USPAT; US-PGPUB; JPO	2004/05/08 14:05
-	1	string same name same items same query same objects	USPAT; US-PGPUB; JPO	2004/05/08 14:05
-	31	string same name same items same query	USPAT; US-PGPUB; JPO	2004/05/08 14:53
-	0	(objects near2 have) same ('more than one' or multiple) same names	USPAT; US-PGPUB; JPO	2004/05/08 15:43
-	0	(objects same have) same ('more than one' or multiple) same names	USPAT; US-PGPUB; JPO	2004/05/08 14:56
-	0	(objects same have) same names	USPAT; US-PGPUB; JPO	2004/05/08 14:56
-	0	(objects same 'have') same names	USPAT; US-PGPUB; JPO	2004/05/08 14:57
-	5451	objects same names	USPAT; US-PGPUB; JPO	2004/05/08 15:17
-	0	(objects same "have") same names	USPAT; US-PGPUB; JPO	2004/05/08 14:57
-	0	objects same names same have	USPAT; US-PGPUB; JPO	2004/05/08 14:57
-	0	objects same names same has	USPAT; US-PGPUB; JPO	2004/05/08 14:57
-	0	object same names same has	USPAT; US-PGPUB; JPO	2004/05/08 14:57
-	0	object\$1 same names same (have or has)	USPAT; US-PGPUB; JPO	2004/05/08 14:58
-	0	object\$1 same name same (have or has)	USPAT; US-PGPUB; JPO	2004/05/08 14:58
-	0	object\$1 same "name" same (have or has)	USPAT; US-PGPUB; JPO	2004/05/08 14:58
-	0	objects same names same "one or more"	USPAT; US-PGPUB; JPO	2004/05/08 15:18

-	2975	object\$1 near3 names	USPAT; US-PGPUB; JPO	2004/05/08 15:19
-	126	object\$1 near3 names same oriented	USPAT; US-PGPUB; JPO	2004/05/12 23:40
-	1	"10/047842"	USPAT; US-PGPUB; JPO	2004/05/08 15:44
-	10	objects same relationship same quer\$ same path	USPAT; US-PGPUB; JPO	2004/05/11 20:47
-	0	objects same relationship same quer\$ same path same second same first	USPAT; US-PGPUB; JPO	2004/05/11 20:48
-	9	name near2 items near2 string	USPAT; US-PGPUB; JPO	2004/05/11 21:31
-	0	objects same "more than one name"	USPAT; US-PGPUB; JPO	2004/05/11 22:16
-	5458	objects same names	USPAT; US-PGPUB; JPO	2004/05/11 22:16
-	0	objects same names same "more than"	USPAT; US-PGPUB; JPO	2004/05/11 22:16
-	512	objects same names same ("one or more" or "a plurality of" or multiple)	USPAT; US-PGPUB; JPO	2004/05/11 22:18
-	14	objects same names same ("one or more" or "a plurality of" or multiple) same aspect\$	USPAT; US-PGPUB; JPO	2004/05/12 14:16
-	7	("6208993" or "5787275" or "5809505" or "6182277" or "5893108").pn. or "2002010840"	USPAT; US-PGPUB; JPO	2004/05/12 10:44
-	0	((("6208993" or "5787275" or "5809505" or "6182277" or "5893108").pn. or "2002010840") and "object names"	USPAT; US-PGPUB; JPO	2004/05/12 10:44
-	804	"object names"	USPAT; US-PGPUB; JPO	2004/05/12 10:45
-	0	"object names" same multiple same enabl\$4	USPAT; US-PGPUB; JPO	2004/05/12 10:45
-	33	"object names" same multiple	USPAT; US-PGPUB; JPO	2004/05/12 10:45
-	14	objects same names same ("one or more" or "a plurality of" or multiple) same aspect\$	USPAT; US-PGPUB; JPO	2004/05/12 14:17
-	1	6182277.pn.	USPAT; US-PGPUB; JPO	2004/05/12 14:35
-	1	5787275.pn.	USPAT; US-PGPUB; JPO	2004/05/12 14:35
-	3	objects same oriented same real same world same query	USPAT; US-PGPUB; JPO	2004/05/12 20:00